

**INFOMATEK**

Volume 18 Nomor 2 Desember 2016

SISTEM PENCEGAHAN SERANGAN BRUTEFORCE PADA UBUNTU SERVER DENGAN MENGGUNAKAN FAIL2BAN

Iwan Kurniawan^{*)}, Ferry Mulyanto, Fuad Nandiasa

Program Studi Teknik Informatika
Fakultas Teknik – Universitas Pasundan

Abstrak: Perkembangan internet yang semakin cepat dengan berbagai macam fungsi dan kebutuhan, menuntut meningkatnya kualitas keamanan jaringan webserver. Terutama dengan semakin terbukanya pengetahuan hacking dan cracking, didukung dengan banyaknya tools yang tersedia dengan mudah dan kebanyakan free, semakin mempermudah para intruder dan attacker untuk melakukan aksi penyusupan ataupun serangan. Masalah timbul ketika sang administrator sedang tidak berada pada posisi siap sedia, misalnya sakit, berada di luar jam kerja, atau adanya kepentingan mendadak, sedangkan serangan terhadap server bisa terjadi kapan saja. Salah satu serangan yang berakibat fatal adalah, serangan bruteforce. Serangan bruteforce pada server memang jarang sekali terjadi, tetapi akibat yang ditimbulkan dari serangan ini adalah penyerang bisa mendapatkan hak akses administrator dan tentu saja membahayakan server. Administrator membutuhkan suatu sistem yang dapat membantu kerjanya. Sebuah sistem yang dapat memberikan hasil laporan apa yang terjadi pada sistem apakah itu sebuah serangan atau penyusupan. Dengan hasil laporan yang di dapat, administrator akan bertindak lebih jauh untuk mencegah terjadinya serangan di masa yang akan datang.

Kata kunci: Bruteforce, Fail2ban

I. PENDAHULUAN

1.1 Latar Belakang

Perkembangan internet yang semakin cepat dengan berbagai macam fungsi dan kebutuhan, menuntut meningkatnya kualitas keamanan jaringan webserver. Terutama dengan semakin terbukanya pengetahuan hacking dan cracking, didukung dengan banyaknya tools yang tersedia dengan mudah dan kebanyakan free, semakin mempermudah para intruder dan attacker untuk melakukan aksi penyusupan ataupun

serangan. Terjadi nya penyusupan atau serangan dapat mengakibatkan masalah pada keberlangsungan sistem yang diserang oleh attacker .

Pencegahaan yang paling sering dilakukan untuk masalah ini adalah dengan menempatkan seorang administrator. Masalah timbul ketika sang administrator sedang tidak berada pada posisi siap sedia, misalnya sakit, berada di luar jam kerja, atau adanya kepentingan mendadak. Sedangkan serangan terhadap server bisa terjadi kapan saja.

^{*)} iwank@unpas.ac.id

Keamanan jaringan komputer mempunyai tiga basis yang harus terpenuhi yang disebut dengan *The Security Trinity*, yaitu *prevention*, *detection* dan *response* (Canavan [1]).

Maka, dari permasalahan tersebut, administrator membutuhkan suatu sistem yang dapat membantu kerjanya. Sebuah sistem yang dapat membantu administrator jika sedang tidak berada di tempat, sebuah sistem yang dapat memberikan hasil laporan apa yang terjadi pada sistem apakah itu sebuah serangan atau penyusupan. Dengan hasil laporan yang di dapat ,administrator akan bertindak lebih jauh untuk mencegah terjadinya serangan di masa yang akan datang.

Di era yang modern ini, teknologi bukan merupakan hal yang baru terutama di perusahaan. Hampir semua perusahaan menggunakan teknologi dalam menjalankan, mengembangkan dan mendukung oprasional segala jenis perusahaan bisnis. Salah satunya persaingan bisnis di bidang peminjaman uang. Perbankan salah satu lembaga intermediasidalam negara, berkenaan hal tersebut bank menuntut para karyawan dapat memberikan pelayanan yang terbaik dan memuaskan kepada nasabah-nasabah. Dari latar belakang di atas, perumusan masalah yakni bagaimana membangun sebuah sistem

yang dapat memberikan notifikasi kepada admin jika terjadi serangan terhadap server.

II. METODOLOGI

Metodologi yang digunakan dalam penelitian ini adalah sebagai berikut:

1. Studi Literatur

Pada tahap ini penulis melakukan pengumpulan fakta-fakta terkait dengan serangan *Bruteforce*

2. Tahap Analisis Lingkungan Penelitian

Tahap ini adalah tahap dimana peneliti melakukan analisis terkait lingkungan penelitian, yaitu *Ubuntu server*.

3. Tahap Pengujian Sebelum Implementasi Fail2ban

Tahap ini adalah tahap dimana dilakukan nya serangan *Bruteforce* terhadap target yang sudah ditentukan yang belum di implementasi Fail2ban. Fail2ban bekerja dengan cara merubah aturan konfigurasi firewall dengan konfigurasi yang berada di Fail2ban itu sendiri, ketika Fail2ban berjalan, ia akan mengambil alih fungsi firewall yang berada di server (Ellingwood [2]).

4. Implementasi Fail2ban

Tahap ini adalah tahap pengimplementasian Fail2ban sebagai

bentuk solusi dari pencegahan serangan *Bruteforce*.

5. Konfigurasi Blocklist.de

Tahap ini adalah tahap dimana *web reporting* di konfigurasi dengan mengisi data dari server yang di implementasi Fail2ban

6. Tahap Pengujian Setelah Implementasi Fail2ban

Tahap ini adalah tahap dimana dilakukan nya pengujian yang sama dengan tahap sebelumnya dengan target yang sudah di implementasi Fail2ban.

7. Analisis Report Blocklist.de

Tahap ini adalah tahap dimana menganalisa hasil *report* yang diterima oleh Blocklist.de dari Ubuntu Server.

default. Untuk *update repository* ketik perintah, *sudo apt-get update*.

2. Fail2ban Package Downloading

Setelah mendapatkan paket *repository* terbaru, langkah selanjutnya adalah mengunduh *Fail2ban*, dan secara otomatis *Fail2ban* akan terpasang pada *Ubuntu server*, dengan mengetikan perintah, *sudo apt-get install fail2ban*.

3. Fail2ban Config File Downloading

Setelah mendapatkan paket *Fail2ban*. Langkah selanjutnya adalah dengan mengunduh *debian configuration file* yang berada di situs *blocklist.de*. *Configuration file* ini berisi pengaturan – pengaturan servis *fail2ban*. Untuk mendapatkan *Configuration file* ketik perintah, *sudo wget <http://www.blocklist.de/downloads/fail2ban.conf.tar.gz>*

4. Fail2ban Config File Extraction

Langkah selanjutnya adalah dengan melakukan extraction pada configuration file tersebut, agar file dapat digunakan oleh aplikasi Fail2ban. Untuk melakukan extraction ketik perintah, *sudo tar -xvzf fail2ban.config.tar.gz*.

5. Fail2ban Config File Replacement

Setelah melakukan *extraction*, *config file* tersebut harus dipindahkan kedalam *folder*

III. IMPLEMENTASI DAN PENGUJIAN

3.1 Instalasi Fail2ban

Fail2ban berfungsi sebagai *monitor* jumlah kegagalan *login ssh* di *server*, yang selanjutnya *ip* akan diblokir. Dalam kasus ini *Fail2ban* menangani serangan bruteforce, untuk itu diperlukan beberapa tahap dalam konfigurasi *Fail2ban*.

1. Update Repository Package

Untuk dapat melakukan instalasi *Fail2ban*, diperlukan *update repository* dikarenakan paket *Fail2ban* tidak tersedia untuk *repository*

dimana *fail2ban* tersebut berada, hal ini diperlukan untuk mempermudah *fail2ban* melakukan logging.

6. Konfigurasi *Blocklist.de*

Blocklist.de web yang menyediakan layanan *fail2ban reporting*. Tujuan digunakan nya *blocklist.de* adalah sebagai *dial-up sender* untuk selanjutnya mengirim hasil *report* beserta detail nya ke *email admin* yang sebelumnya terdaftar di *blocklist.de*. Setelah terdaftar di *blocklist.de*, admin akan diberikan 2 buah *email* yang berperan sebagai *sender* dan *destination mail*. Selanjutnya adalah dengan mendaftarkan *server* yang digunakan

dengan mengisi nama *server*, alamat *IP server*, *sender mail*, dan juga *time-zone*, dimana alamat *IP server* adalah alamat *IP public* dari *server* yang akan digunakan.

7. Konfigurasi *Jail.conf*

Jail.conf adalah file yang berisi konfigurasi – konfigurasi *Fail2ban*, dimana di dalam *jail.conf* kita bisa memilih jenis servis mana yang akan digunakan dan yang tidak digunakan. Dalam kasus ini penulis menggunakan servis *ssh-iptables* dikarenakan percobaan serangan *bruteforce* ini melalui port.

```
enabled = true      menyatakan bahwa fail2ban services di port ssh dijalankan
port    = ssh       port yang dipakai oleh fail2ban services
filter  = sshd       service yang digunakan oleh aksi fail2ban
action  = %(action_mwl)s  action_mwl adalah aksi yang mampu mengirimkan log file ke blocklist.de
maxretry = 6        batas maksimal kegagalan login
```

Gambar 1
Jail.conf

Bisa terlihat dari Gambar 1 bahwa maksimal percobaan login yang dilakukan oleh penyerang adalah 10 kali, jika melebihi 10 kali maka sistem akan memutuskan koneksi antara penyerang dan server, dan setiap percobaan login gagal akan dimasukkan kedalam sebuah log yang terletak di direktori */var/log/auth.log*.

Setelah semua informasi telah dimasukkan kedalam log maka sistem akan mengirimkan sebuah report yang terkait dengan kegiatan penyerangan dengan alamat pengirim *fail2ban[at]dyn.blocklist.de* dan alamat tujuan *fail2ban[at]blocklist.de*.

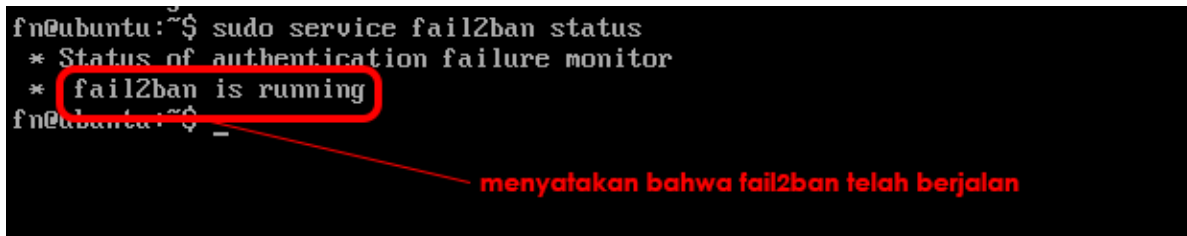
Hal yang selanjutnya dilakukan adalah memilih aksi yang dilakukan oleh *fail2ban* jika terjadi penyerangan. Ada 3 jenis aksi yang

terdapat di dalam *fail.conf*, *Action*, *Action_mw*, dan *Action_mwl*. Dalam kasus ini penulis memilih aksi *Action_mwl*, dikarenakan *Action_mwl* mempunyai kemampuan mengirim email ke alamat yang dituju serta dapat menyimpan semua kegiatan *login* kedalam sebuah *log file*.

8. *Starting and Checking Fail2ban Services*
Starting and Checking Fail2ban Service perlu dilakukan untuk memastikan bahwa *Fail2ban*

berfungsi dengan baik. Untuk melakukan *Starting* ketik perintah *sudo service fail2ban start*.

Setelah melakukan *Starting*, langkah selanjutnya adalah melakukan *Checking* untuk mengetahui bahwa *Fail2ban* telah benar – benar berjalan. Untuk melakukan *Checking* ketik perintah *sudo service fail2ban status*. Bisa dilihat dari Gambar 2 bahwa *Fail2ban* sudah berjalan dengan status *already running*.



```
fn@ubuntu:~$ sudo service fail2ban status
* Status of authentication failure monitor
* fail2ban is running
fn@ubuntu:~$ _
```

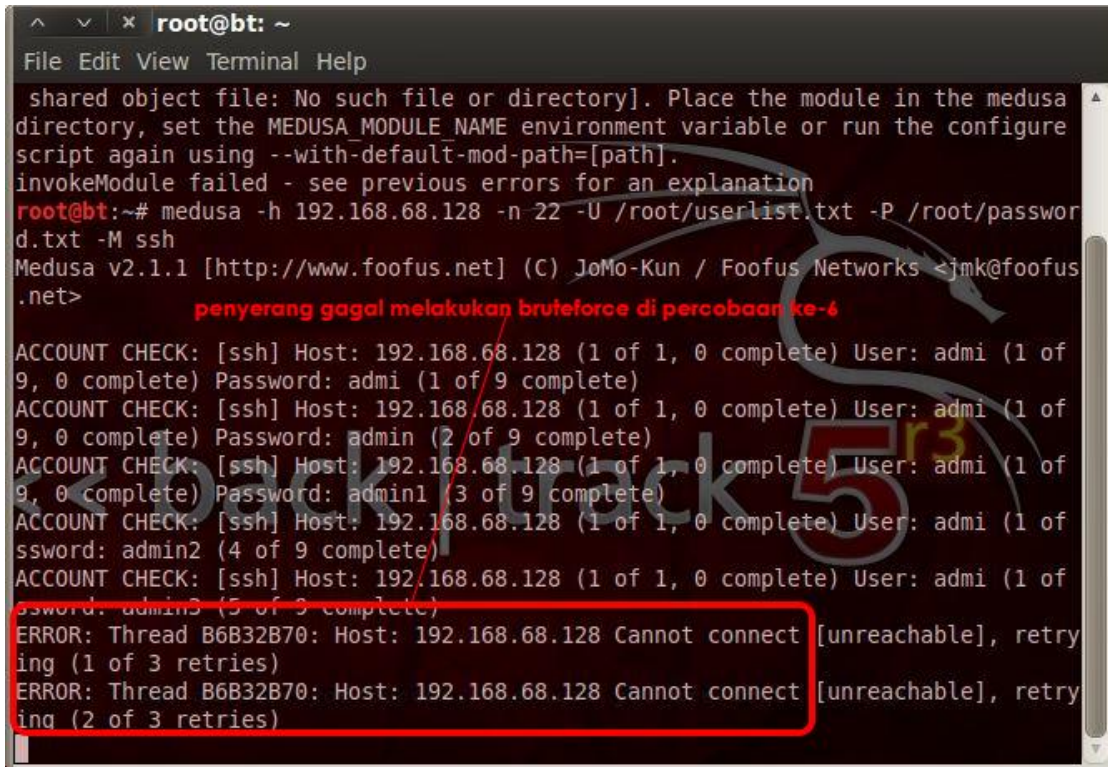
menyatakan bahwa fail2ban telah berjalan

Gambar 2
Starting

3.2 Pengujian Setelah Implementasi Fail2ban

Tahap pengujian ini adalah tahap pengujian untuk melihat dampak serangan *bruteforce* setelah implementasi *fail2ban*. Penyerang menggunakan *tools* yang sama dengan pengujian sebelumnya yaitu dengan menggunakan *medusa*, dengan alamat *ip*

target 192.168.68.128, berbeda dengan alamat *ip target* sebelumnya dikarenakan *ip Ubuntu server* yang digunakan bersifat dinamis. Perintah yang digunakan oleh penyerang pun sama seperti tahap pengujian sebelumnya yaitu, *medusa -h 192.168.68.128 -n 22 -U /root/userlist.txt -P /root/password.txt -M ssh*.



```

root@bt: ~
File Edit View Terminal Help
shared object file: No such file or directory]. Place the module in the medusa
directory, set the MEDUSA_MODULE_NAME environment variable or run the configure
script again using --with-default-mod-path=[path].
invokeModule failed - see previous errors for an explanation
root@bt:~# medusa -h 192.168.68.128 -n 22 -U /root/userlist.txt -P /root/passwor
d.txt -M ssh
Medusa v2.1.1 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus
.net>
penyerang gagal melakukan bruteforce di percobaan ke-6
ACCOUNT CHECK: [ssh] Host: 192.168.68.128 (1 of 1, 0 complete) User: admin (1 of
9, 0 complete) Password: admin (1 of 9 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.68.128 (1 of 1, 0 complete) User: admin (1 of
9, 0 complete) Password: admin1 (2 of 9 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.68.128 (1 of 1, 0 complete) User: admin (1 of
9, 0 complete) Password: admin1 (3 of 9 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.68.128 (1 of 1, 0 complete) User: admin (1 of
9, 0 complete) Password: admin2 (4 of 9 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.68.128 (1 of 1, 0 complete) User: admin (1 of
9, 0 complete) Password: admin3 (5 of 9 complete)
ERROR: Thread B6B32B70: Host: 192.168.68.128 Cannot connect [unreachable], retry
ing (1 of 3 retries)
ERROR: Thread B6B32B70: Host: 192.168.68.128 Cannot connect [unreachable], retry
ing (2 of 3 retries)

```

Gambar 3
Pengujian setelah implemntasi

Terlihat dari Gambar 3 bahwa penyerang tidak berhasil melakukan percobaan *login* yang keenam kali, dimana target otomatis memutuskan koneksi antara penyerang dan sistem. Pemutusan koneksi ini dilakukan oleh *fail2ban* yang terpasang di sistem target dimana penyerang hanya bisa melakukan percobaan *login* sebanyak 6 kali percobaan. Pengaturan terkait dengan maksimal percobaan login sebanyak 6 kali ini disimpan

di dalam *jail.conf* dimana sebelumnya sudah dikonfigurasi. Dengan *Action_mwl* yang sudah ditetapkan sebelumnya sebagai aksi *fail2ban*, maka setiap percobaan gagal login akan disimpan kedalam sebuah *log file* yang terletak di direktori */var/log/auth.log*. Untuk melihat *log file* ketik perintah, *nano /var/log/auth.log*.

```

GNU nano 2.2.6      File: /var/log/auth.log
Nov 1 00:58:00 ubuntu sudo: pam_unix(sudo:session): session opened for user root by fn(uid=0)
Nov 1 00:58:20 ubuntu sudo: pam_unix(sudo:session): session closed for user root
Nov 1 01:01:01 ubuntu sshd[1691]: Invalid user admi from 192.168.68.129
Nov 1 01:01:01 ubuntu sshd[1691]: input_userauth_request: invalid user admi [preauth]
Nov 1 01:01:01 ubuntu sshd[1691]: pam_unix(sshd:auth): check pass; user unknown
Nov 1 01:01:01 ubuntu sshd[1691]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0
Nov 1 01:01:03 ubuntu sshd[1691]: Failed password for invalid user admi from 192.168.68.129 port 49
Nov 1 01:01:03 ubuntu sshd[1691]: pam_unix(sshd:auth): check pass; user unknown
Nov 1 01:01:05 ubuntu sshd[1691]: Failed password for invalid user admi from 192.168.68.129 port 49
Nov 1 01:01:05 ubuntu sshd[1691]: pam_unix(sshd:auth): check pass; user unknown
Nov 1 01:01:07 ubuntu sshd[1691]: Failed password for invalid user admi from 192.168.68.129 port 49
Nov 1 01:01:07 ubuntu sshd[1691]: Disconnecting: Too many authentication failures for admi [preauth]
Nov 1 01:01:07 ubuntu sshd[1691]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=sshd
Nov 1 01:01:18 ubuntu sshd[1693]: Invalid user admi from 192.168.68.129
Nov 1 01:01:18 ubuntu sshd[1693]: input_userauth_request: invalid user admi [preauth]
Nov 1 01:01:18 ubuntu sshd[1693]: pam_unix(sshd:auth): check pass; user unknown
Nov 1 01:01:20 ubuntu sshd[1693]: pam_unix(sshd:auth): check pass; user unknown
Nov 1 01:01:22 ubuntu sshd[1693]: Failed password for invalid user admi from 192.168.68.129 port 49

```

fail2ban memutuskan koneksi dengan penyerang setelah melewati batas maksimal login

Gambar 4

Log file

Bisa terlihat dari Gambar 4 di atas dimana ada sebuah percobaan login yang dilakukan oleh alamat *ip* 192.168.68.129 dimana alamat *ip* tersebut adalah *alamat ip* dari penyerang, dan terlihat bahwa penyerang melakukan percobaan login sebanyak 5 kali yang dinyatakan gagal.

mengirimkan laporan penyerangan ke email melalui third-party applications yaitu blocklist.de, maka seharusnya setiap laporan penyerangan akan dikirimkan terlebih dahulu melalui blocklist.de lalu dilanjutkan ke email admin. Laporan penyerangan bisa dilihat melalui website blocklist.de.

Terkait dengan penjelasan sebelumnya bahwa aksi Action_mwl memiliki kemampuan untuk

Currently, you have entered 1 Server.

id ↑ ↓	name ↑ ↓	IP Address ip ↑ ↓	email ↑ ↓	API-KEY	Activity	Server Quick Links
2470	blocklist report	118.97.186.26	fn.appleid@gmail.com	8775712f84	Attacks: 0 Reports: 0	  

alamat ip server target pengujian

blocklist.de tidak mendapatkan report dari fail2ban

LEGEND:  ATTACK REPORT  API Docs  API URL

Gambar 5

Blocklist de report

Bisa terlihat dari Gambar 5 bahwa alamat *ip* 118.97.186.26 yaitu alamat *ip* server tidak menerima laporan penyerangan ataupun laporan lainnya yang terkait dengan kegiatan di server.

V. KESIMPULAN

Kesimpulan dari penelitian ini solusi yang ditawarkan oleh penulis untuk melakukan implementasi *fail2ban* pada Ubuntu server untuk mencegah serangan *bruteforce*.

1. Implementasi *fail2ban* pada Ubuntu server terbukti dapat mencegah serangan *bruteforce* dan memblokir alamat *ip* dari penyerang tersebut kedalam daftar *blacklist*.

2. *Fail2ban* tidak dapat memberikan *report* kepada administrator melalui web *blocklist.de* dan email administrator.

DAFTAR PUSTAKA

- [1] Canavan, J. E, "*Fundamentals of Network Security*", Artech House, 2001
- [2] Ellingwood, Justin "*How Fail2ban Works to Protect Services on Linux Server*", <https://www.digitalocean.com/community/tutorials/how-fail2ban-works-to-protect-services-on-a-linux-server>, Mei 2014